

## 前回まとめ

- リレーショナルデータベースの意味記述
- 意味: 制約
  - キー(主キー), キー制約
  - 外部キー, 外部キー制約
  - その他の一貫性制約

1

## データベース第4回

第4章 リレーショナルデータモデル  
-- 操作記述 --

2

## リレーショナル代数

- コッドが提案したデータ操作言語
- 理論的な言語システム
- リレーショナル代数そのものをユーザインタフェースに提供しているDBMSはない
- SQLなどの言語のもとになるもの

3

## リレーショナル代数演算

- |          |                    |
|----------|--------------------|
| • 集合演算   | • リレーショナル代数固有の演算   |
| – 和集合演算  | – 射影演算(Projection) |
| – 差集合演算  | – 選択演算(Selection)  |
| – 共通集合演算 | – 結合演算(Join)       |
| – 直積演算   | – 商演算              |

4

## 和両立

リレーション $R(A_1, A_2, \dots, A_n)$ と $S(B_1, B_2, \dots, B_m)$ が和両立とは次の2つの条件を満たしているときをいう.

1.  $R$ と $S$ の**次数が等しい**(つまり  $n=m$ )
2. 各 $i$  ( $1 \leq i \leq n$ )について,  $A_i$ と $B_i$ の**ドメインが等しい**(つまり  $\text{dom}(A_i)=\text{dom}(B_i)$ )

5

## 和両立な2つのリレーション

テニス部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所 属	電 話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

ポイント:

次数が等しい

各ドメインが等しい(ここでは, 列の順番を合わせている)

**属性名は異なっていることもある**

6

## 和集合演算

$R$ と $S$ を和両立なリレーションとする.  $R$ と $S$ に和集合演算を施した結果は,  $R$ と $S$ の和と呼ばれ,  $R \cup S$ で表す. その定義は次の通りである.

$$R \cup S = \{t \mid t \in R \vee t \in S\}$$

7

## 和集合演算するとどうなる?

テニス部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所 属	電 話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員  $\cup$  サッカー部員

8

## 和集合演算するとどうなる？

テニス部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所 属	電 話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員∪サッカー部員

テニス部員∪サッカー部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201
田中桃子	K41	5989-3201

(山田太郎, K55, 5643-3192) なるタプルは重複して現れないことに注意する

「テニス部かサッカー部に所属している人は？」に対応

9

## 差集合演算, 共通集合演算

### • 差集合演算

– RとSを和両立ナリレーションとする. RとSに差集合演算を施した結果は, RとSの差と呼ばれ, R-Sとあらわす. その定義は次の通りである.

$$R - S = \{t | t \in R \wedge \neg(t \in S)\}$$

### • 共通集合演算

– RとSを和両立ナリレーションとする. RとSの共通集合演算を施した結果は, RとSの共通と呼ばれ, R∩Sとあらわす. その定義は次の通りである.

$$R \cap S = \{t | t \in R \wedge t \in S\}$$

10

## 差集合演算は？

テニス部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所 属	電 話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員 − サッカー部員

11

## 差集合演算は？

テニス部員

氏 名	所 属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所 属	電 話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員 − サッカー部員

テニス部員 − サッカー部員

氏 名	所 属	連絡先
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

「テニス部にしか所属していない人は？」に対応

12

## 共通集合演算は？

テニス部員

氏名	所属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所属	電話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員∩サッカー部員

13

## 共通集合演算は？

テニス部員

氏名	所属	連絡先
山田太郎	K55	5643-3192
鈴木花子	K41	5591-0585
佐藤一郎	K55	5274-5201

サッカー部員

部員名	所属	電話
田中桃子	K41	5989-3201
山田太郎	K55	5643-3192

テニス部員∩サッカー部員

テニス部員∩サッカー部員

氏名	所属	連絡先
山田太郎	K55	5643-3192

「テニス部とサッカー部の両方に所属している人は？」に対応

14

## 直積演算

$R(A_1, A_2, \dots, A_n)$ と $S(B_1, B_2, \dots, B_m)$ を2つのリレーションとする。このとき  $R$  と  $S$  の直積、これを  $R \times S$  と書く、は次のように定義される  $n+m$  次のリレーションである。

$$R \times S = \{(r, s) \mid r \in R \wedge s \in S\}$$

15

社員

社員番号	社員名	給与	所属
0650	山田太郎	50	K55
1508	鈴木花子	40	K41
0231	田中桃子	60	K41
2034	佐藤一郎	40	K55

(a) リレーション 社員

部門

部門番号	部門名	部門長
K55	データベース	0650
K41	ネットワーク	1508

(b) リレーション 部門

社員×部門

社員 社員番号	社員 社員名	社員 給与	社員 所属	部門 部門番号	部門 部門名	部門 部門長
0650	山田太郎	50	K55	K55	データベース	0650
0650	山田太郎	50	K55	K41	ネットワーク	1508
1508	鈴木花子	40	K41	K55	データベース	0650
1508	鈴木花子	40	K41	K41	ネットワーク	1508
0231	田中桃子	60	K41	K55	データベース	0650
0231	田中桃子	60	K41	K41	ネットワーク	1508
2034	佐藤一郎	40	K55	K55	データベース	0650
2034	佐藤一郎	40	K55	K41	ネットワーク	1508

(c) 社員×部門

16

## 直積演算の例

従業員

従業員番号	部門番号	氏名	年齢
001	1	尾下 真樹	27
002	2	下戸 彩	17
004	1	宇田 ヒカル	20

部門

部門番号	部門名
1	開発
2	営業

×

=

従業員. 従業員番号	従業員. 部門番号	従業員.氏名	従業員. 年齢	部門. 部門番号	部門. 部門名
001	1	尾下 真樹	27	1	開発
002	2	下戸 彩	17	1	開発
004	1	宇田 ヒカル	20	1	開発
001	1	尾下 真樹	27	2	営業
002	2	下戸 彩	17	2	営業
004	1	宇田 ヒカル	20	2	営業

17

## 直積演算とは？

R

A1	A2	A3
a	b	c
d	a	e
a	d	c

S

B1	B2	B3
b	f	a
d	a	e

18

## 直積演算結果

R

A1	A2	A3
a	b	c
d	a	e
a	d	c

S

B1	B2	B3
b	f	a
d	a	e

RXS

A1	A2	A3	B1	B2	B3
a	b	c	b	f	a
d	a	e	b	f	a
a	d	c	b	f	a
a	b	c	d	a	e
d	a	e	d	a	e
a	d	c	d	a	e

19

## リレーショナル代数演算

- 集合演算
  - 和集合演算
  - 差集合演算
  - 共通集合演算
  - 直積演算
- リレーショナル代数固有の演算
  - 射影演算(Projection)
  - 選択演算(Selection)
  - 結合演算(Join)
  - 商演算

20

## 射影演算(Projection)

$R(A_1, A_2, \dots, A_n)$ をリレーション,  $R$ の全属性集合  $\{A_1, A_2, \dots, A_n\}$ の部分集合を  $X=\{A_{i_1}, A_{i_2}, \dots, A_{i_k}\}$ , ここに  $1 \leq i_1 < i_2 < \dots < i_k \leq n$  とする. このとき  $R$ の  $X$ 上の射影, これを  $R[X]$ (あるいは  $R[A_{i_1}, A_{i_2}, \dots, A_{i_k}]$ )と書く, は次のように定義されるリレーションである.

$$R[X] = \{t[X] \mid t \in R\}$$

21

## 射影演算結果

供給			供給[供給元, 部品]	
供給元	部品	供給先	供給元	部品
A11	P101	K55	A11	P101
A11	P102	K51	A11	P102
A11	P102	K41	A12	P102
A12	P102	K41	A12	P103
A12	P103	K51	A13	P101
A13	P101	K41	A13	P102
A13	P102	K51	A13	P103
A13	P103	K51		

(a) リレーション 供給

(b) リレーション 供給の属性集合 {供給元, 部品} 上の射影

(A11, P102) なるタプルは重複して現れないことに注意する

ポイント:

タプルの個数が変化している

これは, ケースバイケース(どんな時かはわかりますか?)

22

## 射影演算

- リレーション  $R(A_1, A_2, \dots, A_n)$  がもつ属性のうち指定したもののみを残し, 他を取り除く単項演算
- $\{A_1', A_2', \dots, A_m'\} \subseteq \{A_1, A_2, \dots, A_n\}$  のとき, 射影  $\Pi_{A_1', A_2', \dots, A_m'}(R)$  は次のように定義される

$$\Pi_{A_1', A_2', \dots, A_m'}(R) = \{t[A_1', A_2', \dots, A_m'] \mid t \in R\}$$

- $t[A_1', A_2', \dots, A_m']$  はタプル  $t$  から属性  $A_1', A_2', \dots, A_m'$  の値のみ残し, 他の属性値を削除したタプル

23

## 選択演算(Selection)

$R(A_1, A_2, \dots, A_n)$ をリレーション,  $A_i$ と $A_j$ を $\theta$ -比較可能な属性とする. このとき,  $R$ の $A_i$ と $A_j$ 上の $\theta$ -選択, これを  $R[A_i \theta A_j]$ と書く, は次のように定義されるリレーションである.

$$R[A_i \theta A_j] = \{t \mid t \in R \wedge t[A_i] \theta t[A_j]\}$$

$A_i$ と $A_j$ が $\theta$ -比較可能とは,  $\text{dom}(A_i) = \text{dom}(A_j)$ かつ  $R$ の任意のタプル  $t$  に対して,  $t$  を変数とする述語  $t[A_i] \theta t[A_j]$  の真か偽が常に定まるときをいう.

$\theta: >, \geq, =, \leq, <, \neq$

24

## 大なり選択演算結果

商品

商品番号	商品名	原価	売価	定価
G110	刺身	600	500	980
G120	豆腐	90	75	120
G130	卵	95	100	140
G140	コーヒー豆	700	860	860
G150	ケーキ	200	250	300

(a) リレーション 商品 (定休日前日の夕方5時過ぎのスーパーマーケットを想定)

商品[原価>売価]

商品番号	商品名	原価	売価	定価
G110	刺身	600	500	980
G120	豆腐	90	75	120

(b) リレーション 商品の属性 原価 と 売価 上の大なり選択

25

## 選択演算

- リレーション  $R(A_1, A_2, \dots, A_n)$  がもつタプルのうち、指定した条件を満たすタプルのみを残し、他のタプルを削除する単項演算

$$\sigma_F(R) = \{t \mid t \in R \wedge P_F(t)\}$$

- $P_F(t)$  はタプル  $t$  が選択条件  $F$  を満足するとき真となる述語である
- 選択条件  $F$ 
  - (1)  $A_i \theta C$   $A_i$  は属性値,  $C$  は定数,  $\theta$  は比較演算子 ( $=, \leq, >, <, \geq, \neq$ )
  - (2)  $A_i \theta A_j$   $A_i$  と  $A_j$  のドメインは同一であること (比較できる)
  - (3) (1), (2) の条件を論理和 ( $\vee$ ), 論理積 ( $\wedge$ ), 否定 ( $\neg$ ) を用いて組み合わせたもの

26

## 射影演算・選択演算の例

従業員

従業員番号	部門番号	氏名	年齢
001	1	尾下 真樹	27
002	2	下戸 彩	17
003	3	本村 拓哉	30
004	1	宇田 ヒカル	20

$\pi$  氏名, 年齢 (従業員)

氏名	年齢
尾下 真樹	27
下戸 彩	17
本村 拓哉	30
宇田 ヒカル	20

$\sigma$  年齢 > 20 (従業員)

従業員番号	部門番号	氏名	年齢
001	1	尾下 真樹	27
003	3	本村 拓哉	30

選択は必要なタプル(行)のみを取り出す

射影は必要な属性(列)のみを取り出す

27

## 射影演算と選択演算

$R$

A1	A2	A3
a	b	c
d	a	e
a	d	c

$\pi_{A1, A3}(R)$

A1	A3
a	c
d	e

$\sigma_F(R)$

A1	A2	A3
a	b	c
a	d	c

$F: A3=c$

28

## 結合演算

$R(A_1, A_2, \dots, A_n)$ と $S(B_1, B_2, \dots, B_m)$ を2つのリレーション、 $A_i$ と $B_j$ を $\theta$ -比較可能とする。このとき $R$ と $S$ の $A_i$ と $B_j$ 上の $\theta$ -結合、これを $R[A_i \theta B_j]S$ と書く、は次のように定義される。

$$R[A_i \theta B_j]S = \{(t, u) \mid t \in R \wedge u \in S \wedge t[A_i] \theta u[B_j]\}$$

社員[所属=部門番号]部門

社員.社員番号	社員.社員名	社員.給与	社員.所属	部門.部門番号	部門.部門名	部門.部門長
0650	山田太郎	50	K55	K55	データベース	0650
1508	鈴木花子	40	K41	K41	ネットワーク	1508
0231	田中桃子	60	K41	K41	ネットワーク	1508
2034	佐藤一郎	40	K55	K55	データベース	0650

29

## 結合演算

結合条件 $F$ をリレーション $R$ の属性 $A_i$ とリレーション $S$ の属性 $B_j$ の比較演算子 $\theta$ による比較条件 $A_i \theta B_j$ とするとき、結合は以下のリレーションを導出する

$$R \bowtie_F S = \{t \times u \mid t \in R \wedge u \in S \wedge P_F(t, u)\}$$

$P_F(t, u)$ は $t$ と $u$ が条件 $F$ を満足するとき真となる述語

$$R \bowtie_F S = \sigma_F(R \times S) \quad \theta \text{結合, } \theta \text{が} = \text{のとき等結合}$$

30

## 結合演算

R

A	B
1	3
2	5
3	4

S

C	D	E
4	5	3
4	4	6
5	5	7

等結合

$$R \bowtie_{B=C} S$$

A	B	C	D	E
2	5	5	5	7
3	4	4	5	3
3	4	4	4	6

属性B, Cは比較可能  
(ドメインが同じ)

A	B	C	D	E
2	5	4	5	3
2	5	4	4	6

$\theta$ 結合

$$R \bowtie_{B > C} S$$

31

## 等結合の例

従業員

従業員番号	部門番号	氏名	年齢
001	1	尾下 真樹	27
002	2	下戸 彩	17
004	1	宇田 ヒカル	20

部門

部門番号	部門名
1	開発
2	営業

$$\bowtie_{\text{部門番号} = \text{部門番号}}$$

=

例のように複数のリレーションの情報を組み合わせるために、結合が使われる

従業員.従業員番号	従業員.部門番号	従業員.氏名	従業員.年齢	部門.部門番号	部門.部門名
001	1	尾下 真樹	27	1	開発
004	1	宇田 ヒカル	20	1	開発
002	2	下戸 彩	17	2	営業

結合は直積と選択の組合せで計算される(次スライド)

32



## 等結合の例(1.直積演算)

従業員				×	部門	
従業員番号	部門番号	氏名	年齢		部門番号	部門名
001	1	尾下 真樹	27		1	開発
002	2	下戸 彩	17		2	営業
004	1	宇田 ヒカル	20			

  

従業員 従業員番号	従業員 部門番号	従業員.氏名	従業員 年齢	部門 部門番号	部門 部門名
001	1	尾下 真樹	27	1	開発
002	2	下戸 彩	17	1	開発
004	1	宇田 ヒカル	20	1	開発
001	1	尾下 真樹	27	2	営業
002	2	下戸 彩	17	2	営業
004	1	宇田 ヒカル	20	2	営業

33

## 等結合の例(2.選択演算)

従業員 従業員番号	従業員 部門番号	従業員.氏名	従業員 年齢	部門 部門番号	部門 部門名
001	1	尾下 真樹	27	1	開発
002	2	下戸 彩	17	1	開発
004	1	宇田 ヒカル	20	1	開発
001	1	尾下 真樹	27	2	営業
002	2	下戸 彩	17	2	営業
004	1	宇田 ヒカル	20	2	営業

  

$\sigma_{\text{従業員.部門番号}=\text{部門.部門番号}}$

例のように複数の  
リレーションの  
情報を組み合わ  
せるために、結  
合が使われる

従業員 従業員番号	従業員 部門番号	従業員.氏名	従業員 年齢	部門 部門番号	部門 部門名
001	1	尾下 真樹	27	1	開発
004	1	宇田 ヒカル	20	1	開発
002	2	下戸 彩	17	2	営業

## 自然結合

- 自然結合 (natural join)

$$R * S = \pi_{A_1, \dots, A_n, B_1, \dots, B_m, C_1, \dots, C_k} \left( \sigma_{R.B_1=S.B_1 \wedge \dots \wedge R.B_m=S.B_m} (R \times S) \right)$$

$$R(A_1, \dots, A_n, B_1, \dots, B_m), S(B_1, \dots, B_m, C_1, \dots, C_k)$$

$$= R \bowtie S$$

- 実際の応用でよく使われる重要な演算子
- 2つのリレーションを同一の属性同士で等結合し、結合結果から同一の属性を取り除いたもの
  - 等結合の結果には、同一の属性値を持つ属性が重複して存在することになり無駄

35

## 自然結合の例

従業員番号	部門番号	氏名	年齢	部門番号	部門名
001	1	尾下 真樹	27	1	開発
002	2	下戸 彩	17	2	営業
004	1	宇田 ヒカル	20		

部門番号の同じタプル同士の組合せになる

従業員番号	部門番号	氏名	年齢	部門名
001	1	尾下 真樹	27	開発
002	2	下戸 彩	17	営業
004	1	宇田 ヒカル	20	開発

※ 各属性のものとのリレーション名は省略できる

36

## 商演算

$R(A_1, A_2, \dots, A_{n-m}, B_1, B_2, \dots, B_m)$ を $n$ 次,  $S(B_1, B_2, \dots, B_m)$ を $m$ 次( $m < n$ )のリレーションとする.  
 $R$ を $S$ で割った商, これを $R \div S$ と書く, は次のように定義されるリレーションである.

$$R \div S = \{t \mid t \in R[A_1, A_2, \dots, A_{n-m}] \wedge (\forall u \in S)((t, u) \in R)\}$$

37

## 商演算

受講	
学生	科目
田中	データベース
田中	プログラム
田中	人工知能
佐藤	人工知能
佐藤	ネットワーク
辻	データベース

必須	
必須	
	プログラム
	データベース
	人工知能

受講÷必須	
学生	
田中	

「必須科目をすべて受講している学生は？」に対応

38

## 商演算

全種類の部品を購入する顧客？

顧客

顧客名	購入部品名
C1	B
C2	B
C2	N
C3	N
C4	N
C4	B

部品

部品名
N
B

商

顧客名
C2
C4

39

## 商演算

全種類の部品を購入する顧客？

顧客

顧客名	購入部品名
C1	B
C2	B
C2	N
C3	N
C4	N
C4	B

部品

部品名
N
B

商

顧客名
C2
C4

40

## 8つの演算は独立ではない

- たとえば  $R \cap S = R - (R - S)$

41

## リレーショナル代数表現

1. リレーショナルデータベースの実リレーションRは表現である。
2. RとSを表現とすると、RとSが和両立なら、 $R \cup S$ 、 $R - S$ 、 $R \cap S$ は表現である。
3. RとSを表現とすると、 $R \times S$ は表現である。
4. Rを表現とすると、 $R[X]$ は表現である。ここに、XはRの属性のなす集合である。
5. Rを表現とすると、 $R[A_i \theta A_j]$ は表現である。ここに、 $A_i$ と $A_j$ はRの属性で $\theta$ -比較可能とする。
6. RとSを表現とすると、 $R[A_i \theta B_j]S$ は表現である。ここに、 $A_i$ と $B_j$ はそれぞれRとSの属性で、 $\theta$ -比較可能とする。
7. RとSを表現とすると、 $R \div S$ は表現である。
8. 以上の定義によって得られた表現のみがリレーショナル代数表現である。

42

## リレーショナル代数表現

- 8つのリレーショナル代数演算の結果はリレーションになる
- リレーショナル代数演算を施した結果に新たにリレーショナル代数演算を施すことができる

43

## リレーショナル代数表現の演習

- 4.3の演習は各自しっかり勉強すること
- ポイント
  - リレーショナル代数演算を理解し、それぞれどういう結果を得るものかを覚える
  - 問合せは、「すべての学生を求めよ」のように、自然言語で表現される。それをリレーショナル代数演算で表現できるようになる

44

## 空とその意味

- 記載する属性値がないとき, 空(null)を使う
- 空の主な3種類
  - unk : unknown 未知の  
例) 配偶者はいるけどそのデータが提供されてない
  - dne : nonexistent 存在しない  
例) 配偶者がいない
  - ni : no-information 情報がない  
例) 配偶者がいるかいないかわからない

45

## 空とリレーショナル代数演算の拡張

供給				需要			
供給元	部品	供給元	部品	供給元	部品	供給元	部品
A11	P101	A11	P101	P101	K55	P101	K55
A13	P103	A13	P103	P102	K41	P102	K41

  

供給 [部品=部品] 需要				供給 ⋈ <sub>部品=部品</sub> 需要			
供給	供給元	供給	部品	供給	供給元	供給	部品
A11	P101	P101	K55	A11	P101	P101	K55
A13	P103	—	—	A13	P103	—	—

(a) 内結合

  

供給 ⋈ <sub>部品=部品</sub> 需要				供給 ⋈ <sub>部品=部品</sub> 需要			
供給	供給元	供給	部品	供給	供給元	供給	部品
A11	P101	P101	K55	A11	P101	P101	K55
—	—	P102	K41	A13	P103	—	—

(c) 右外結合

  

供給 ⋈ <sub>部品=部品</sub> 需要				供給 ⋈ <sub>部品=部品</sub> 需要			
供給	供給元	供給	部品	供給	供給元	供給	部品
A11	P101	P101	K55	A11	P101	P101	K55
A13	P103	—	—	A13	P103	—	—
—	—	—	—	—	—	P102	K41

(d) 完全外結合

46

## まとめ: リレーショナル代数演算

- 集合演算
  - 和集合演算
  - 差集合演算
  - 共通集合演算
  - 直積演算
- リレーショナル代数固有の演算
  - 射影演算(Projection)
  - 選択演算(Selection)
  - 結合演算(Join)
  - 商演算

47

## リレーショナル代数演算子のまとめ

- 基本的な演算
  - 2項演算
    - 和, 差, 直積
  - 単項演算
    - 射影, 選択
- その他の演算
  - 2項演算
    - 結合, 自然結合, 共通部分, 商

48

## リレーショナル代数演算子の主な用途

- 選択(S), 射影(P)
  - 必要なデータ(表の行)や属性(表の列)を取り出すために使用
- 結合(J)
  - 複数のリレーションを組み合わせるために使用
- 和, 差, 共通部分
  - 複数の演算結果同士を組み合わせるときに使用
- 直積
  - 直接は使用しない(結合を定義する上で重要)
- 商(あまり使わない)